IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellant: Dale C. Morris, et al. | Examiner: Midys Rojas

Serial No.: 09/499,720 | Group Art Unit: 2185

Filed: February 8, 2000 | Docket No.: 10991915-1

Title: PRIVILEGE PROMOTION BASED ON CHECK OF PREVIOUS PRIVILEGE LEVEL

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir/Madam:

This Appeal Brief is submitted in support of the Notice of Appeal filed December 20, 2007, appealing the rejection of claims 1-24 of the above-identified application as set forth in the Final Office Action mailed October 29, 2007.

An Appeal Brief was previously filed on August 7, 2006, and a corresponding Appeal Brief fee of $500.00 was paid. Accordingly, per MPEP 1207.4, only the difference between the current fee of $510.00 for filing a Brief in Support of an Appeal as set forth under 37 C.F.R. § 41.20(b)(2) and the previously paid $500.00 Appeal Brief fee needs to be paid.

The U.S. Patent and Trademark Office is hereby authorized to charge **Deposit Account No. 08-2025** in the amount of **$10.00** for filing a Brief in Support of an Appeal as set forth under 37 C.F.R. § 41.20(b)(2). At any time during the pendency of this application, please charge any required fees or credit any overpayment to Deposit Account No. 08-2025.

Appellants respectfully request consideration and reversal of the Examiner's rejection of pending claims 1-24.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The real party in interest is Hewlett-Packard Development Company, LP having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

RELATED APPEALS AND INTERFERENCES

Appellants submit that there are no related appeals or interferences that will directly affect or be directly affected by or have a bearing on the Board's decision in the present appeal.

STATUS OF CLAIMS

In a Final Office Action mailed October 29, 2007, claims 1-24 were finally rejected. Claims 1-24 are pending in the application, and are the subject of the present Appeal.

STATUS OF AMENDMENTS

No amendments have been entered subsequent to the Final Office Action mailed October 29, 2007. The claims listed in the Claims Appendix, therefore, reflect the claims as of October 29, 2007.

SUMMARY OF THE CLAIMED SUBJECT MATTER

The subject matter of the independent claims involved in the Appeal is related to a method of promoting a current privilege level of a processor of a computer system controlled by an operating system, wherein the current privilege level controls application instruction execution in the computer system by controlling accessibility to system resources.

One aspect of the present invention, as claimed in independent claim 1, provides a method (100) of promoting a current privilege level (52) of a processor (32) of a computer system (30) controlled by an operating system (36), wherein the current privilege level (52) controls application instruction (56) execution in the computer system (30) by controlling

accessibility to system resources. The method (100) includes performing a privilege promotion instruction (62) by the operating system (36). The privilege promotion instruction (62) is stored in a first page (58) of memory (34) not writeable by application instructions (56) at a first privilege level. The privilege promotion instruction (62) includes reading a stored previous privilege level state (70) and comparing the read previous privilege level state (70) to the current privilege level (52). If the previous privilege level state (70) is equal to or less privileged than the current privilege level (52), the current privilege level (52) is promoted to a second privilege level which is higher than the first privilege level. *See Specification*, page 5, line 25 – page 10, line 19; and Figures 1 and 2.

Another aspect of the present invention, as claimed in independent claim 6, provides a method (100) of executing instructions in a computer system (30) controlled by an operating system (36). The method (100) includes executing application instructions (56) in a processor (32) of the computer system (30) at a current privilege level (52) of the processor (32) equal to a first privilege level. The application instructions (56) are stored in a first page (54) of memory (34). The current privilege level (52) controls application instruction (56) execution in the computer system (30) by controlling accessibility to system resources. The method (100) includes performing a call instruction (104) to a second page (58) of memory (34) not writeable by the application instructions (56) at the first privilege level. The call instruction (104) includes storing a return address (106) to the first page (54) of memory (34), storing the first privilege level in a previous privilege level state (70), and performing a privilege promotion instruction (62) by the operating system (36). The privilege promotion instruction (62) is stored in the second page (58) of memory (34). The privilege promotion instruction (62) includes reading the stored previous privilege level state (70) and comparing the read previous privilege level state (70) to the current privilege level (52). If the previous privilege level state (70) is equal to or less privileged than the current privilege level (52), the current privilege level (52) is promoted to a second privilege level which is higher than the first privilege level. *See Specification*, page 5, line 25 – page 10, line 19; and Figures 1 and 2.

Yet another aspect of the present invention, as claimed in independent claim 12, provides a computer system (30). The computer system (30) includes a processor (32)

having a current privilege level (52) which controls application instruction (56) execution in the computer system (30) by controlling accessibility to system resources and having a previous privilege level state (70). The computer system (30) includes a memory (34) having a plurality of memory pages (54, 58) including a first memory page (58) storing a privilege promotion instruction (62). The first memory page (58) is not writeable by application instructions (56) at a first privilege level. The computer system (30) includes an operating system (36) stored in the memory (34) for controlling the processor (32) and memory (34) and for performing the privilege promotion instruction (62). The privilege promotion instruction (62) reads the previous privilege level state (70) and compares the read previous privilege level state (70) to the current privilege level (52). If the previous privilege level state (70) is equal to or less privileged than the current privilege level (52), the current privilege level (52) is promoted to a second privilege level which is higher than the first privilege level. *See Specification*, page 5, line 25 – page 10, line 19; and Figures 1 and 2.

Yet another aspect of the present invention, as claimed in independent claim 17, provides a computer system (30). The computer system (30) includes a processor (32) having a current privilege level (52) which controls application instruction (56) execution in the computer system (30) by controlling accessibility to system resources. The computer system (30) includes a memory (34) having a plurality of memory pages (54, 58) including a first memory page (54) storing application instructions (56) and a second memory page (58) storing a higher privileged routine (60) and a privilege promotion instruction (62). The second memory page (58) is not writeable by the application instructions (56) at a first privilege level. The computer system (30) includes an operating system (36) stored in the memory (34) for controlling the processor (32) and memory (34). The processor (32) executes the application instructions (56) with the current privilege level (52) equal to the first privilege level and the application instructions (56) perform a call instruction (104) to the second memory page (58). The call instruction (104) stores a return address (106) to the first memory page (54) and stores the first privilege level in a previous privilege level state (70). The operating system (36) performs the privilege promotion instruction (62). The privilege promotion instruction (62) reads the stored previous privilege level state (70) and compares the read previous privilege level state (70) to the current privilege level (52). If the previous

privilege level state (70) is equal to or less privileged than the current privilege level (52), the current privilege level (52) is promoted to a second privilege level which is higher than the first privilege level. *See Specification*, page 5, line 25 – page 10, line 19; and Figures 1 and 2.

Yet another aspect of the present invention, as claimed in independent claim 23, provides a computer readable medium containing a privilege promotion instruction (62) for controlling a computer system (30) to perform a method (100) of promoting a current privilege level (52) of a processor (32) of the computer system (30). The current privilege level (52) controls application instruction (56) execution in the computer system (30) by controlling accessibility to system resources. The method (100) includes reading a stored previous privilege level state (70) and comparing the read previous privilege level state (70) to the current privilege level (52). If the previous privilege level state (70) is equal to or less privileged than the current privilege level (52), the current privilege level (52) is promoted to a privilege level which is higher than the current privilege level. *See Specification*, page 5, line 25 – page 10, line 19; and Figures 1 and 2.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Whether claims 1-24 are patentable under 35 U.S.C. § 103(a) over the Arora U.S. Patent No. 6,393,556 in view of the Banning et al. U.S. Patent No. 6,363,336.

ARGUMENT

I. The Applicable Law

With regard to a 35 U.S.C. § 103 obviousness rejection: "Patent examiners carry the responsibility of making sure that the standard of patentability enunciated by the Supreme Court and by the Congress is applied in each and every case." M.P.E.P. 2141 (emphasis in the original). The Examiner bears the burden under 35 U.S.C. § 103 in establishing a *prima facie* case of obviousness. *In re Fine*, 837 F.2d 1071, 1074 [5 USPQ2d 1596, 1598] (Fed. Cir. 1988).

One criteria that must be satisfied to establish a *prima facie* case of obviousness is the reference or combined references must teach or suggest all of the claim limitations. *In re Royka*, 490 F.2d 981 [180 USPQ 580] (C.C.P.A. 1974).

However, "[a] patent composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art." *KSR Int'l Co. v. Teleflex, Inc.,* 127 S. Ct. 1727, 1731 [82 USPQ2d 1385, 1389] (2007). In making an obviousness determination over a combination of prior art references, it is "important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does." *Id. at* 1738 [1396].

In order to facilitate review of the determination of whether there was an apparent reason to combine known elements in the fashion claimed by the patent at issue, the "analysis should be made explicit." *Id. at* 1738 [1396]. "[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *In re Kahn*, 441 F.3d 977, 988 [78 USPQ2d 1329] (Fed. Cir. 2006) (cited with approval in *KSR*, 127 S. Ct. at 1738 [82 USPQ2d at 1396])

The test for obviousness under § 103 must take into consideration the invention as a whole; that is, one must consider the particular problem solved by the combination of elements that define the invention. *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143 [227 USPQ 543, 551] (Fed. Cir. 1985). Furthermore, claims must be interpreted in light of the specification, claim language, other claims, and prosecution history. *Panduit Corp. v. Dennison Mfg. Co.*, 810 F.2d 1561, 1568 [1 USPQ2d 1593, 1597] (Fed. Cir. 1987), *cert. denied*, 481 U.S. 1052 (1987). At the same time, a prior patent cited as a § 103 reference must be considered in its entirety, "*i.e.* as a *whole*, including portions that lead away from the invention." *Id.* That is, the Examiner must recognize and consider not only the similarities, but also the critical differences between the claimed invention and the prior art as one of the factual inquiries pertinent to any obviousness inquiry under 35 U.S.C. § 103. *In re Bond*, 910 F.2d 831, 834 [15 USPQ2d 1566, 1568] (Fed. Cir. 1990) (emphasis added).

Furthermore, the Examiner must avoid hindsight. *Id.* "A fact finder should be aware, of course, of the distortion caused by hindsight bias and must be cautious of arguments reliant

upon *ex post* reasoning." *KSR,* 127 S. Ct. at 1739 [82 USPQ2d at 1397] (citing to *Graham v. John Deere*, 383 U.S. 1 [148 USPQ 459] (1966) in warning against a temptation to read into the prior art the teachings of the invention at issue and instructing courts to guard against slipping into the use of hindsight).

"[W]hen the prior art teaches away from combining certain known elements, discovery of a successful means of combining them is more likely to be nonobvious." *KSR,* 127 S. Ct. at 1737 [82 USPQ2d at 1395] (citing to *United States v. Adams*, 383 U.S. 39, 51-52 [148 USPQ 479] (1966).

In conclusion, an Appellant is entitled to a patent grant if a *prima facie* case of obviousness is not established. The Federal Circuit has endorsed this view in stating: "If examination at the initial stage does not produce a *prima facie* case of unpatentability, then without more the Appellant is entitled to grant of the patent." In re Oetiker, 977 F.2d 1443, 1446 [24 USPQ2d 1443, 1448] (Fed. Cir. 1992).

II. Rejection of Claims 1-24 under 35 U.S.C. § 103(a) as being unpatentable over the Arora U.S. Patent No. 6,393,556 in view of the Banning et al. U.S. Patent No. 6,363,336.

The combination of the Arora Patent and the Banning Patent fail to render claims 1-24 *prima facie* obvious. Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, fail to teach or suggest the limitations recited by independent claims 1, 6, 12, 17, and 23.

The Arora Patent discloses changing a privilege level in a processor configured to pipeline instructions. The processor includes a first memory storing an architectural privilege level that is set at a first privilege level, a second memory storing a plurality of instructions, and a pipeline including a plurality of processing stages. A first instruction is fetched from the memory and a determination is made whether the first instruction requires the first privilege level be changed to a second privilege level, and in response thereto, any subsequent instructions are flushed from the pipeline before recording the second privilege level in the first memory. (Abstract).

In the Arora Patent, the processor 30 maintains a "current privilege level" (CPL) 38 in a memory storage device. The CPL is maintained in the processor's register set. The operating system sets the CPL to prevent the user from performing dangerous or insecure operations. If the pipeline 30 is currently processing an application program instruction, a prior instruction would have set the CPL 38 to the proper privilege level. If an instruction requiring a higher privilege level follows the current instruction, an instruction, such as an "enter privilege code" (EPC) instruction, that directs the processor to change the privilege level of the CPL must first be processed to increase the privilege level. (Col. 4, lines 13-27).

In the Arora Patent, after decoding an instruction directing the processor to change the CPL 38 from a first to a second privilege level, the processor compares the second privilege level to the CPL 38. (Col. 6, lines 27-31). The processor will compare the CPL 38 with the privilege level specified in the EPC instruction. If the EPC instruction directs the processor to change the CPL 38 to a higher privilege level, the processor flushes any instructions in the pipeline subsequent to the EPC instruction, and continues processing the EPC instruction. When the EPC instruction is retired, the CPL 38 privilege level is increased. If the EPC instruction specifies a privilege level lower than or the same as the CPL 38, the processor will issue a fault. (Col. 6, lines 46-59).

The Banning Patent discloses a method for determining if writes to a memory page are directed to target instructions which have been translated to host instructions in a computer which translates instructions for a target instruction set to a host instruction set, including the steps of detecting a write to a memory page storing target instructions which have been translated to host instructions, detecting whether a sub-area of the memory page to which the write is addressed stores target instructions which have been translated, and invalidating host instructions translated from addressed target instructions. (Abstract).

Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, fail to teach or suggest the limitations recited by independent claim 1 including **performing a privilege promotion instruction by the operating system, the privilege promotion instruction being stored in a first page of memory not writeable by application instructions at a first privilege level, the privilege promotion instruction including: reading a stored previous privilege level state; comparing the read previous**

privilege level state to the current privilege level; and if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a second privilege level which is higher than the first privilege level.

The Examiner submits that the Arora Patent discloses the privilege promotion instruction being stored in a memory (instruction memory 36 storing a plurality of instructions… see Figure 2). (Final Office Action mailed October 29, 2007, page 7). The Examiner admits that the Arora Patent does not teach the instruction memory 36 including a page of memory not writeable by application instructions at a first privilege level. The Examiner submits that the Banning Patent discloses an instruction memory where memory pages marked with a T bit are protected and if a write is attempted, an exception is generated. The Examiner further submits that in this system, the T bit provides the protection so that the memory page is not writeable at the first privilege level. (Final Office Action mailed October 29, 2007, page 8).

Instruction memory 36 of Figure 2 of the Arora Patent is a cache memory for storing instructions that are processed in the pipeline 32. A cache memory is a portion of memory that operates faster than main memory. The first time an instruction is executed, it must be loaded from the relatively slow main memory. Recently-accessed memory locations are saved in the cache in case they are needed again, so each instruction will be saved in the cache after being loaded from the memory the first time. (Col. 3, lines 24-35). Instruction memory 36 does not include a first page of memory not writeable by application instructions at a first privilege level. Instruction memory 36 is a cache memory where all instructions are stored for execution. The Arora Patent discloses that a typical pipeline 32 includes a stage during which instructions are fetched from the memory 36. (Col. 3, lines 44-46). The EPC instruction 111 is fetched from the instruction memory 36. (Col. 4, lines 50-51).

The Banning Patent also fails to teach or suggest a *privilege promotion instruction* being stored in a first page of memory not writeable by application instructions at a first privilege level. In addition, there is no teaching or suggestion for one skilled in the art to combine the cache memory of the Arora Patent with the T bit system of the Banning Patent or to replace the cache memory of the Arora Patent with the T bit system of the Banning Patent. Cache memories store data temporarily for faster access, therefore one skilled in the

art would not implement the T bit system of the Banning Patent to protect instructions temporarily stored in the cache memory of the Arora Patent from being overwritten. In addition, one skilled in the art would not replace the cache memory of the Arora Patent with the T bit system of the Banning Patent because the T bit system would not provide the fast access to the memory as required by the Arora Patent. Therefore, the Arora Patent and the Banning Patent fail to teach or suggest *the privilege promotion instruction being stored in a first page of memory not writeable by application instructions at a first privilege level.*

The Examiner submits that the claim 1 limitation of reading a stored previous privilege level state is disclosed by the Arora Patent at column 4, lines 19-22 by register CPL 38, which stores the privilege level set by a previous instruction. The Examiner also submits that comparing the read previous privilege level state to the current privilege level is disclosed in column 6, lines 46-49 of the Arora Patent. Further, the Examiner also submits that the privilege level stored in CPL 38 is the previous privilege level since it represents a previous instruction, while the privilege level related to the EPC is the current privilege level since it represents the current instruction. (Final Office Action mailed October 29, 2007, page 7). In addition, the Examiner submits that "at the moment of comparison, the privilege level of EPC is the privilege level necessary for the instruction that is currently being prepared for execution in the system (instruction requiring a higher priority level follows in the pipeline), thus it is a current privilege level. Also, at the moment of comparison, the CPL is the previous privilege level because it was the privilege level set by a prior instruction (Col. 4, lines 19-28), and it is the privilege level that was necessary for the execution of an instruction that was executed previous to the instruction corresponding to the EPC." (Final Office Action mailed October 29, 2007, page 4).

The Arora Patent states, however, that "[i]f an instruction requiring a higher privilege level follows the *current* instruction, an instruction, such as an 'enter privilege code' ('EPC') instruction, that directs the processor to change the privilege level of the architectural CPL must first be processed to increase the privilege level." (Col. 4, lines 22-27). The privilege level stored in CPL 38 is not updated until after the EPC instruction is executed. The privilege level stored in CPL 38 remains the current privilege level during the execution of the EPC instruction.

The cited text of the Arora Patent discloses comparing the architectural CPL with the privilege level specified in the EPC instruction. (Col. 6, lines 46-49). The Arora Patent fails to disclose the claim 1 limitations of *reading a stored previous privilege level state* and *comparing the read previous privilege level state to the current privilege level*. In the Arora Patent, a previous privilege level state is not stored and therefore cannot be read. The privilege level stored in CPL 38 is never the previous privilege level as submitted by the Examiner. CPL 38 remains the current privilege level until an EPC instruction is retired.

The Arora Patent also discloses that the CPL is compared to the privilege level specified in the EPC instruction. In contrast, claim 1 requires comparing the *read previous privilege level state* to the *current privilege level*. The privilege level of the EPC instruction does not teach or suggest the current privilege level. Rather, the EPC instruction directs the processor to change the privilege level of the CPL. (Col. 4, lines 24-26). The EPC instruction provides a potential future privilege level, not the current privilege level. The Arora Patent discloses that an EPC instruction *eventually may* cause the processor 30 to change the architectural CPL 38 to a second privilege level. (Col. 4, lines 56-58).

The Examiner states that the claim 1 limitations of "if a previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a second privilege level which is higher than the first privilege level" are disclosed by the Arora Patent "since the EPC instruction directs the processor to change the architectural privilege level to a higher privilege level . . .". (". . . increase the architectural privilege level from privilege level 3 to privilege level 0"). The Examiner further states that in comparing privilege levels, the stored privilege level (stored in CPL 38) must be read in the comparison process. (Final Office Action mailed October 29, 2007, page 7).

The Arora Patent discloses raising the current architectural CPL to the privilege level specified in the EPC instruction. (Col. 6, lines 46-49). The Arora Patent, however, does not disclose the claim 1 limitation of promoting the current privilege level to a second privilege level which is higher than the first privilege level if the previous privilege level state is equal to or less privileged than the current privilege level.

Claim 1 recites if the *previous privilege level state* is *equal* to or *less privileged* than the *current privilege level*, promoting the current privilege level. In contrast, the Examiner

states that the Arora Patent discloses increasing the *current privilege level* if the privilege level of the *CPL* is *lower* than the privilege level of the *EPC*. The Arora Patent discloses if the EPC instruction specifies a privilege level lower than, *or the same as*, the architectural CPL, the processor will issue a fault rather than promote the CPL. (Col. 6, lines 56-59). In addition, the Examiner submits that the privilege level of the EPC instruction discloses the current privilege level state and the CPL discloses the stored previous privilege level state. (Final Office Action mailed October 29, 2007, page 4). Further, the Examiner submits that the privilege level of the EPC instruction is the privilege level necessary for the instruction that is currently being prepared for execution in the system, thus it is a current privilege level; and that the CPL is the previous privilege level because it was the privilege level necessary for the execution of a previous instruction. (Final Office Action mailed October 29, 2007, page 4). Therefore, based on the Examiner's interpretation, claim 1 would recite if the CPL is equal to or less privileged than the privilege level of the EPC instruction, promoting the current privilege level. The CPL, however, is the current privilege level, not the previous privilege level state, and the privilege level of the EPC instruction is a future privilege level, not the current privilege level as submitted by the Examiner. The CPL in the Arora Patent is not increased to the privilege level of the EPC instruction until the EPC instruction is retired. (Col. 6, lines 54-56).

In addition, the Examiner states that "regardless of what the privilege levels are labeled as by the Arora Patent, the examiner is free to interpret the true meaning and relationship of these privilege levels within the system of Arora." (Final Office Action mailed October 29, 2007, page 3). As outlined above, the labels given to the privilege levels by the Arora Patent are not merely labels, but define the meaning and function of the privilege levels and the relationship between the privilege levels. The Examiner is not just relabeling the privilege levels within the system of the Arora patent to match the limitations of claim 1, but rather the Examiner is improperly redefining the privilege levels within the system of the Arora Patent to match the limitations of claim 1.

In view of the above, Appellants believe independent claim 1 to be allowable over the Arora Patent and the Banning Patent. Dependent claims 2-5 further define patentably distinct

independent claim 1. Accordingly, dependent claims 2-5 are also believed to be allowable over the Arora Patent and the Banning Patent.

For similar reasons as discussed above with reference to independent claim 1, and for additional reasons discussed below, Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest the limitations recited by independent claim 6 including **performing a call instruction to a second page of memory not writeable by the application instructions at the first privilege level, the call instruction including: storing a return address to the first page of memory; and storing the first privilege level in a previous privilege level state; and performing a privilege promotion instruction by the operating system, the privilege promotion instruction being stored in the second page of memory, the privilege promotion instruction including: reading the stored previous privilege level state; comparing the read previous privilege level state to the current privilege level; and if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a second privilege level which is higher than the first privilege level.**

In addition to the reasons discussed above with reference to independent claim 1, the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest *a call instruction including storing a return address to the first page of memory*. It appears that the Examiner has failed to directly address this claim limitation.

Further, the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest *a call instruction including storing the first privilege level in a previous privilege level state*. The Examiner submits that the CPL 38 is the previous privilege level state and the privilege level of the EPC instruction is the current privilege level. (Final Office Action mailed October 29, 2007, page 4). The Examiner also submits that the privilege level of the EPC instruction is the privilege level necessary for the instruction that is currently being prepared for execution in the system, thus it is a current privilege level; and that the CPL is the previous privilege level because it was the privilege level necessary for the execution of a previous instruction. (Final Office Action mailed October 29, 2007, page 4).

The Arora Patent does not disclose a call instruction that stores the first privilege level in a previous privilege level state. Even if CPL 38 stores the previous privilege level as suggested by the Examiner, there is no call instruction that stores the privilege level in CPL 38. Based on the Examiner's interpretation, CPL 38 merely becomes the stored previous privilege level state when the EPC instruction is executed. In addition, claim 6, recites *executing application instructions in a processor of the computer system at a current privilege level of the processor equal to a first privilege level, and wherein the current privilege level controls application instruction execution in the computer system by controlling accessibility to system resources*. The current privilege level as recited in claim 6 is not the privilege level of previous instructions but rather the privilege level for currently executing instructions. Further, in the Arora Patent the CPL is never stored in a previous privilege level state. The CPL is increased when an EPC instruction is retired. (Col. 6, lines 54-56). The Arora Patent discloses a current privilege level and a privilege level of an EPC instruction (which is a *future privilege level*) as opposed to the current privilege level and the stored *previous privilege level* state as recited by claim 6.

In view of the above, Appellants believe independent claim 6 to be allowable over the Arora Patent and the Banning Patent. Dependent claims 7-11 further define patentably distinct independent claim 6. Accordingly, dependent claims 7-11 are also believed to be allowable over the Arora Patent and the Banning Patent.

For similar reasons as discussed above with reference to independent claim 1, Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest the limitations recited by independent claim 12 including **a memory having a plurality of memory pages including a first memory page storing a privilege promotion instruction, wherein the first memory page is not writeable by application instructions at a first privilege level;** and **performing the privilege promotion instruction as follows: reads the previous privilege level state; compares the read previous privilege level state to the current privilege level; and if the previous privilege level state is equal to or less privileged than the current privilege level, promotes the current privilege level to a second privilege level which is higher than the first privilege level**.

In view of the above, Appellants believe independent claim 12 to be allowable over the Arora Patent and the Banning Patent. Dependent claims 13-16 further define patentably distinct independent claim 12. Accordingly, dependent claims 13-16 are also believed to be allowable over the Arora Patent and the Banning Patent.

For similar reasons as discussed above with reference to independent claim 1, Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest the limitations recited by independent claim 17 including **a memory having a plurality of memory pages including a first memory page storing application instructions and a second memory page storing a higher privileged routine and a privilege promotion instruction, wherein the second memory page is not writeable by the application instructions at a first privilege level; wherein the processor executes the application instructions with the current privilege level equal to the first privilege level and the application instructions perform a call instruction to the second memory page as follows: stores a return address to the first memory page; and stores the first privilege level in a previous privilege level state; and wherein the operating system performs the privilege promotion instruction as follows: reads the stored previous privilege level state; compares the read previous privilege level state to the current privilege level; and if the previous privilege level state is equal to or less privileged than the current privilege level, promotes the current privilege level to a second privilege level which is higher than the first privilege level**.

In view of the above, Appellants believe independent claim 17 to be allowable over the Arora Patent and the Banning Patent. Dependent claims 18-22 further define patentably distinct independent claim 17. Accordingly, dependent claims 18-22 are also believed to be allowable over the Arora Patent and the Banning Patent.

For similar reasons as discussed above with reference to independent claims 1 and 6, Appellants submit that the Arora Patent and the Banning Patent, either alone, or in combination, also fail to teach or suggest the limitations recited by independent claim 23 including a computer readable medium containing a privilege promotion instruction for controlling a computer system to perform a method including **reading a stored previous privilege level state; comparing the read previous privilege level state to the current**

privilege level; and if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a privilege level which is higher than the current privilege level.

In view of the above, Appellants believe independent claim 23 to be allowable over the Arora Patent and the Banning Patent. Dependent claim 24 further defines patentably distinct independent claim 23. Accordingly, dependent claim 24 is also believed to be allowable over the Arora Patent and the Banning Patent.

In view of the above, Appellants respectfully request reversal of the rejection of claims 1- 24 under 35 U.S.C. § 103(a).

CONCLUSION

For the above reasons, Appellants respectfully submit that the cited references neither anticipate nor render obvious the pending claims of the Present Application. The pending claims distinguish over the cited references, and therefore, Appellants respectfully submit that the rejections must be withdrawn, and respectfully request the Examiner be reversed and claims 1-24 be allowed.

Any inquiry regarding this Appeal Brief should be directed to either David A. Pettner at Telephone No. (408) 447-3013, Facsimile No. (408) 447-0854 or Patrick G. Billig at Telephone No. (612) 573-2003, Facsimile No. (612) 573-2005. In addition, all correspondence should continue to be directed to the following address:

IP Administration
Legal Department, M/S 35
HEWLETT-PACKARD COMPANY
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Respectfully submitted,

Dale C. Morris, et al.,

By,

DICKE, BILLIG & CZAJA, PLLC
Fifth Street Towers, Suite 2250
100 South Fifth Street
Minneapolis, MN 55402
Telephone: (612) 573-2003
Facsimile: (612) 573-2005

Date: February 20, 2008
PGB:kmh

/Patrick G. Billig/
Patrick G. Billig
Reg. No. 38,080

CLAIMS APPENDIX

1. (Original) A method of promoting a current privilege level of a processor of a computer system controlled by an operating system, wherein the current privilege level controls application instruction execution in the computer system by controlling accessibility to system resources, the method comprising:

performing a privilege promotion instruction by the operating system, the privilege promotion instruction being stored in a first page of memory not writeable by application instructions at a first privilege level, the privilege promotion instruction including:
reading a stored previous privilege level state;

comparing the read previous privilege level state to the current privilege level; and

if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a second privilege level which is higher than the first privilege level.

2. (Original) The method of claim 1 wherein the step of performing the privilege promotion instruction further includes:

if the previous privilege level state is more privileged than the current privilege level, taking an illegal operation fault.

3. (Original) The method of claim 1 wherein the system resources include system registers.

4. (Original) The method of claim 1 wherein the system resources include system instructions.

5. (Original) The method of claim 1 wherein the system resources include memory pages.

6. (Original) A method of executing instructions in a computer system controlled by an operating system, the method comprising:

executing application instructions in a processor of the computer system at a current privilege level of the processor equal to a first privilege level, wherein the application instructions are stored in a first page of memory, and wherein the current privilege level controls application instruction execution in the computer system by controlling accessibility to system resources;

performing a call instruction to a second page of memory not writeable by the application instructions at the first privilege level, the call instruction including:

storing a return address to the first page of memory; and

storing the first privilege level in a previous privilege level state; and

performing a privilege promotion instruction by the operating system, the privilege promotion instruction being stored in the second page of memory, the privilege promotion instruction including:

reading the stored previous privilege level state;

comparing the read previous privilege level state to the current privilege level; and

if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a second privilege level which is higher than the first privilege level.

7. (Original) The method of claim 6 further comprising:

performing a return instruction including:

transferring instruction control flow to the stored return address to the first page of memory; and

demoting the current privilege level to the stored previous privilege level state.

8. (Original) The method of claim 6 wherein the step of performing the privilege promotion instruction further includes:

if the previous privilege level state is more privileged than the current privilege level, taking an illegal operation fault.

9. (Original) The method of claim 6 wherein the system resources include system registers.

10. (Original) The method of claim 6 wherein the system resources include system instructions.

11. (Original) The method of claim 6 wherein the system resources include memory pages.

12. (Original) A computer system comprising:

a processor having a current privilege level which controls application instruction execution in the computer system by controlling accessibility to system resources and having a previous privilege level state;

a memory having a plurality of memory pages including a first memory page storing a privilege promotion instruction, wherein the first memory page is not writeable by application instructions at a first privilege level; and

an operating system stored in the memory for controlling the processor and memory, and performing the privilege promotion instruction as follows:

reads the previous privilege level state;

compares the read previous privilege level state to the current privilege level; and

if the previous privilege level state is equal to or less privileged than the current privilege level, promotes the current privilege level to a second privilege level which is higher than the first privilege level.

13. (Original) The computer system of claim 12 wherein the operating system performing the privilege promotion instruction further includes:

if the previous privilege level state is more privileged than the current privilege level, taking an illegal operation fault.

14. (Original) The computer system of claim 12 further comprising:

system registers, and wherein the system resources include the system registers.

15. (Original) The computer system of claim 12 wherein the system resources include system instructions.

16. (Original) The computer system of claim 12 wherein the system resources include memory pages.

17. (Original) A computer system comprising:

a processor having a current privilege level which controls application instruction execution in the computer system by controlling accessibility to system resources;

a memory having a plurality of memory pages including a first memory page storing application instructions and a second memory page storing a higher privileged routine and a privilege promotion instruction, wherein the second memory page is not writeable by the application instructions at a first privilege level;

an operating system stored in the memory for controlling the processor and memory;

wherein the processor executes the application instructions with the current privilege level equal to the first privilege level and the application instructions perform a call instruction to the second memory page as follows:

stores a return address to the first memory page; and

stores the first privilege level in a previous privilege level state; and

wherein the operating system performs the privilege promotion instruction as follows:

reads the stored previous privilege level state;

compares the read previous privilege level state to the current privilege level; and

if the previous privilege level state is equal to or less privileged than the current privilege level, promotes the current privilege level to a second privilege level which is higher than the first privilege level.

18. (Original) The computer system of claim 17 wherein the processor via the higher privileged routine performs a return instruction as follows:

transfers instruction control flow to the stored return address to the first page of memory; and

demotes the current privilege level to the stored previous privilege level state.

19. (Original) The computer system of claim 17 wherein the operating system performing the privilege promotion instruction further includes:

if the previous privilege level state is more privileged than the current privilege level, taking an illegal operation fault.

20. (Original) The computer system of claim 17 further comprising:

system registers, and wherein the system resources include the system registers.

21. (Original) The computer system of claim 17 wherein the system resources include system instructions.

22. (Original) The computer system of claim 17 wherein the system resources include memory pages.

23. (Original) A computer readable medium containing a privilege promotion instruction for controlling a computer system to perform a method of promoting a current privilege level of a processor of the computer system, wherein the current privilege level controls application instruction execution in the computer system by controlling accessibility to system resources, the method comprising:

reading a stored previous privilege level state;

comparing the read previous privilege level state to the current privilege level; and

if the previous privilege level state is equal to or less privileged than the current privilege level, promoting the current privilege level to a privilege level which is higher than the current privilege level.

24. (Original) The computer readable medium of claim 23 wherein the method of promoting the current privilege level further comprises:

if the previous privilege level state is more privileged than the current privilege level, taking an illegal operation fault.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.